CHAPTER 17

Using Inline Elements and Special Characters

In This Chapter

An introduction to inline elements

Choosing between logical and physical inline elements

Web character sets

Adding foreign languages

Adding inline quotes

Grouping inline elements with the SPAN element

Inline elements enable you to specify the structure of your document further. They also enable you to add emphasis to specific words in a way style sheets can't. This chapter takes you through the details of inline elements and special characters. It includes information about using foreign languages, adding inline quotes, and using the SPAN element to change the presentation of a group of elements within a block.

Introducing Inline Elements

Inline elements are the parts of your document that fill the block elements. You may have a paragraph of text defined with the P element (a block-level element). Within the P element, you can define inline elements, such as quotes, book titles (to be italicized), words in a foreign language, words to which you'd like to draw attention (to be bolded), and special characters (such as a copyright symbol).

Inline elements give you the control to make these specific formatting changes within your block elements. Table 17-1 includes all inline elements.

Table 17-1
All Inline Elements

Element	*Description*	*Deprecated*	*Logical or Physical*
ABBR	abbreviated form		L
ACRONYM	instance of an acronym		L
B	bold		P
BIG	big		P
CITE	citation		L
CODE	computer code fragment		L
DEL	deleted text		L
DFN	instance definition		L
EM	emphasis		L
I	italics		P
INS	inserted text		L
KBD	text to be entered by the user		L
Q	quotation		L
S	strikeout	deprecated (use DEL)	P
SAMP	sample of code		L
SMALL	small		P
SPAN	container for grouping within a block element		
STRIKE	strikeout	deprecated (use DEL)	P
STRONG	strong		L
SUB	subscript		P
SUP	superscript		P
TT	teletype		P
U	underline	Deprecated (use DEL)	P
VAR	variable or program argument		L

All inline elements have a few things in common. They all require both start and end tags. They can all have other inline elements as part of their content. They can all be grouped for purposes of formatting with style sheets with the SPAN element.

You can nest inline elements. All nestings are not valid. For instance, nesting a CODE element within a CITE element gives you unpredictable results. Even more important with inline elements than with block elements is that you close the elements in the reverse order they were opened.

Logical Versus Physical Styles

There is more than one way to italicize text as it appears in your browser. You can use the straight-shootin' approach and use the I element, you can use the more indirect route and use the EM element, or you can use the CITE element because the phrase you want italicized is a book title. What is the difference? It depends on whether you are using italics for the sake of using italics or whether you are using italics to convey some other information.

This is a philosophical discussion, really. Purists, from the SGML school, believe all text formatting is associated with some deeper meaning. You don't bold because you want bold, you bold because you want the words you think should be bolded to come across as stronger than the rest of the text. As it turns out, most browsers render the STRONG element the same as the BOLD element.

It doesn't reflect poorly on you either way. You can make your own call as to whether you use the physical inline elements or the logical inline elements. Table 17-2 shows physical styles; Table 17-3 shows logical styles.

Table 17-2
Physical Inline Elements

Element	*Description*	*Renders As*
B	bold	bold
I	italics	italics
S	strikeout	text with a line through it
SMALL	small	smaller than the regular font
STRIKE	strikeout	text with a line through it
SUB	subscript	text that falls below the baseline of the rest of the text
SUP	superscript	text whose baseline is above the baseline of the rest of the text
TT	teletype	fixed-width font
U	underline	underlined text (obsoleted: don't use this; it is too confusing because links are underlined)

Table 17-3
Logical Inline Elements

Element	*Description*	*Renders As*
ABBR	abbreviated form	nothing special
ACRONYM	acronym	nothing special
CITE	citation	italics (I)
CODE	code sample	fixed-width font (TT)
DEL	deleted text	text with a line through it (STRIKE or S); renders differently in IE and Netscape
DFN	definition	italics (I); renders differently in IE and Netscape
EM	emphasis	italics (I)
INS	inserted text	underlined (U); renders differently in IE and Netscape
KBD	code sample	fixed-width font (TT)
Q	quote	nothing special
SAMP	code sample	fixed-width font (TT)
STRONG	strong	bold (B)
VAR	program variable	italics (I)

Understanding Web Character Sets

The Web supports several character sets. This includes the standard English character set, the extended English character set (which includes all types of accents for the vowels, mathematical symbols, reference symbols, and Greek letters), and internationalization characters. To include characters from any of these character sets, select the appropriate entity from Appendix D.

Adding Special Characters

What if you need to insert the copyright symbol? Use an entity. Entities can be inserted into inline elements or block elements. Remember to use an ampersand (&) at the beginning of the entity and a semicolon (;) at the end. The most common special characters are included in Table 17-4. A complete list of entities is available in Appendix D.

Table 17-4
Common Special Character Entities

Character Notation	*Hex Notation*	*Entity Created*
nbsp		nonbreaking space
iexcl	¡	inverted exclamation mark
cent	¢	cent sign
pound	£	pound sign
curren	¤	currency sign
yen	¥	yen sign
sect	§	section sign
copy	©	copyright sign
laquo	«	left-pointing, double-angle quotation mark
not	¬	not sign
shy		discretionary hyphen
reg	®	registered trademark sign
deg	°	degree sign
plusmn	±	plus-or-minus sign
para	¶	paragraph sign
raquo	»	right-pointing, double-angle quotation mark
frac14	¼	fraction one-quarter
frac12	½	fraction one-half
frac34	¾	fraction three-quarters
iquest	¿	inverted question mark

Special Characters for Specific Jobs

If you are doing anything with math, you should be able to display the mathematical symbols. The most common of these are in Table 17-5.

Table 17-5
Common Mathematical Entities

Character Notation	*Hex Notation*	*Entity Created*
forall	∀	for all
part	∂	partial differential
exist	∃	there exists
empty	∅	empty set = null set = diameter
nabla	∇	nabla = backward difference
isin	∈	element of
notin	∉	not an element of
Ni	∋	contains as member
prod	∏	n-ary product = product sign
sum	∑	n-ary summation
minus	−	minus sign
lowast	∗	asterisk operator
radic	√	square root = radical sign
prop	∝	proportional to
infin	∞	infinity
ang	∠	angle
and	∧	logical and = wedge
Or	∨	logical or = vee
cap	∩	intersection = cap
cup	∪	union = cup
int	∫	integral
there4	∴	therefore
sim	∼	tilde operator = varies with = similar to
cong	≅	approximately equal to
asymp	≈	almost equal to = asymptotic to
Ne	≠	not equal to
equiv	≡	identical to
Le	≤	less than or equal to
ge	≥	greater than or equal to

Character Notation	*Hex Notation*	*Entity Created*
sub	⊂	subset of
sup	⊃	superset of
nsub	⊄	not a subset of
sube	⊆	subset of or equal to
supe	⊇	superset of or equal to
oplus	⊕	circled plus = direct sum
otimes	⊗	circled times = vector product
perp	⊥	up tack = orthogonal to = perpendicular
sdot	⋅	dot operator
lceil	⌈	left ceiling
rceil	⌉	right ceiling
lfloor	⌊	left floor
rfloor	⌋	right floor
lang	〈	left-pointing angle bracket
rang	〉	right-pointing angle bracket

How about if you need special punctuation symbols? See Table 17-6 for the most common entities that represent punctuation.

Table 17-6
Common Punctuation Entities

Character Notation	*Hex Notation*	*Entity Created*
bull	•	bullet = black small circle
hellip	…	horizontal ellipsis = three-dot leader
prime	′	prime = minutes = feet
Prime	″	double prime = seconds = inches
oline	‾	overline = spacing overscore
frasl	⁄	fraction slash

Handling Foreign Languages

What if you need to include text from other languages? Greek? French (with its accents)? Or even Chinese? The first thing you must do is to define the base language for your page. You can do this in the HTML element itself. Your HTML element would look like this:

```
<HTML lang="EN">
```

The value of the lang attribute should be a valid two-character language code. If you then have a paragraph with a different language as the base — say Spanish — then you could identify the base language for that paragraph by putting a lang attribute into the P element like this:

```
<P lang="SP">
```

Sometimes you simply want to include a sentence or a word in another language. The SPAN element is most appropriate for those times. A complete list of language codes is in Appendix F.

There is a lot more to showing foreign languages that require a different alphabet on your page. If you have only a sentence or two to display, your best bet is probably to use an image of the text. If you have a considerable amount of text to display in a language other than English — say Japanese — which uses Kanjii and Katakana, your visitors must run a Japanese operating system because Japanese, Chinese, and other character-based languages use a double-byte character set, quite different from the single-byte character set (ASCII) used for English. For more information about this complex topic, visit www.samsung.co.kr. You may want to click on the English tab in the upper right corner.

You probably noticed in Chapter 15 that most of the block-level elements include both lang and dir attributes. These often need to work together. Some languages are actually read right to left. HTML accommodates the creation of pages using these languages by rendering in the direction you dictate. You still create the text in a left-to-right direction, but it is rendered right-to-left, if you indicate, thus

```
<BODY lang="CH" dir="rtl">
```

would give you a body with a base language of Chinese and a text direction of right to left. In Chapter 12, an element called BDO was introduced as a new element. This is the bidirectional algorithm override element. It enables you to override the base direction. The dir attribute is mandatory for the BDO element.

Bidirectional Algorithm Override <BDO>

Start Tag: Required

Content: Inline elements

End Tag: Required

Attributes: dir: Mandatory: ltr or rtl

id, class: used in conjunction with style sheets

lang: language

style: for style sheets

title: gives the element a title

Adding Quotes: The Q Element

The Q element is a prospective element. The Q element doesn't render as anything other than ordinary text, but it may someday in the future. The Q element is the inline equivalent of BLOCKQUOTE and takes the same attributes.

Quote <Q>

Start Tag: Required

Content: Inline elements

End Tag: Required

Attributes: id, class: used in conjunction with style sheets

lang, dir: indicate language and text direction

cite: URL of the quote if taken from the Web

title: title of this element

style: define style here to override that in a style sheet

events: see Chapter 48

Grouping Inline Elements: The SPAN Element

The SPAN element is the inline equivalent of the DIR element. You can group any number of elements within a block-level element or between block-level elements by enclosing them in a SPAN element.

The SPAN element can then use an id or a class, as defined in a style sheet, to change the look of the elements it encompasses. Suppose you want to change the background to yellow for part of some text (to make it look like it is highlighted). You can use the SPAN element as follows:

```
<P>It is very important to proofread <SPAN class="highlight">
all your text </SPAN> carefully. You can find examples of <SPAN
class="highlight">common misspellings in Table A4.</SPAN> As
Strunk and White always say, <SPAN class="highlight"><Q>"You
should always find time to proofread."</Q> Spell checking
simply is not enough.</SPAN></P>
```

This text would render as follows. The yellow background text is shown as underlined.

```
It is important to proofread all your text carefully. You can
find examples of common misspellings in Table A4. As Strunk and
White always say, "You should always find time to proofread."
Spell checking simply is not enough.
```

And the style sheet would look like this:

```
SPAN.highlight {
              background: yellow}
```

From Here

Go to Chapter 26 and learn the CSS syntax.

Jump to Chapter 48 and learn about events and JavaScript.

Proceed to Chapter 18 and learn about hyperlinks.

Summary

This chapter covered the essentials of inline elements. You learned the difference between physical inline elements, which actually make the browser render in a certain way, and logical inline elements, which tell the browser what message you are trying to send with your text. You also learned about inline elements that currently don't render differently at all, such as ABBR, ACRONYM, and Q.

This chapter explained the use of entities in depth, including using them to display mathematical symbols, punctuation, and foreign languages. You learned about text language and direction. Finally, you learned about the SPAN element, which enables you to apply a style across inline elements.

✦ ✦ ✦